# 3DET-Mamba: State Space Model for End-to-End 3D Object Detection

**Mingsheng Li**[1,*], **Jiakang Yuan**[1,*], **Sijin Chen**[1,2], **Lin Zhang**[1],
**Anyu Zhu**[1], **Xin Chen**[2], **Tao Chen**[1,†]
[1] Fudan University [2] Tecent PCG
[*] Equal contribution [†] Corresponding author

## Abstract

Transformer-based architectures have been proven successful in detecting 3D objects from point clouds. However, the quadratic complexity of the attention mechanism struggles to encode rich information as point cloud resolution increases. Recently, state space models (SSM) such as Mamba have gained great attention due to their linear complexity and long sequence modeling ability for language understanding. To exploit the potential of Mamba on 3D scene-level perception, for the first time, we propose 3DET-Mamba, which is a novel SSM-based model designed for indoor 3D object detection. Specifically, we divide the point cloud into different patches and use a lightweight yet effective Inner Mamba to capture local geometric information. To observe the scene from a global perspective, we introduce a novel Dual Mamba module that models the point cloud in terms of spatial distribution and continuity. Additionally, we design a Query-aware Mamba module that decodes context features into object sets under the guidance of learnable queries. Extensive experiments demonstrate that 3DET-Mamba surpasses previous 3DETR on indoor 3D detection benchmarks such as ScanNet, improving AP@0.25/AP@0.50 from 65.0%/47.0% to 70.4%/54.4%, respectively.

## 1 Introduction

The aim of 3D object detection [24, 43, 26, 35] from point clouds is to locate and recognize objects present in 3D scenes. It is a challenging task since point clouds are often irregular, sparse, and unordered. To directly work with point clouds, VoteNet [31] utilizes PointNet++ [33] to extract features from irregular point clouds, which are then fed into a decoder to generate the 3D bounding boxes. Motivated by the success of Transformer [40] in computer vision [8, 34, 20, 3, 19], some works [24, 14, 26, 4] try to design Transformer-based 3D detectors. 3DETR [26] proposes an end-to-end transformer-based architecture to generate bounding boxes from raw point clouds. However, with limited computational resources, the quadratic complexity of the attention mechanism struggles to encode detailed representations, as it relies on increasing the point cloud resolution (*i.e.*, longer point cloud sequences).

Recently, state space models (SSMs) [10, 45, 16] have received significant attention due to their linear complexity and long-sequence modeling ability. As Mamba [10] demonstrates a strong ability to handle long sequences in natural language processing, it has rapidly been employed on different tasks (*e.g.*, image and 3D object classification [23, 60, 21, 22], video understanding [18, 1] and motion generation [55, 46]) and achieves great success. This motivates us to take advantage of Mamba in capturing long-range dependencies to extract more detailed representations in complex 3D scenes.

However, directly integrating Mamba [10] into the off-the-shelf detectors cannot achieve satisfactory results on 3D object detection tasks due to the following challenges. Firstly, SSMs like Mamba are causal models designed for handling 1-D sequence data, making it difficult to model unordered and

38th Conference on Neural Information Processing Systems (NeurIPS 2024).


https://doi.org/10.52202/079017-1497

non-causal 3-D point clouds. Secondly, the original Mamba block focuses on modeling long-range global information but lacks the ability to extract local features, which are essential for point cloud learning [25, 29]. Besides, previous works primarily use Mamba as an encoder for single object analysis, it remains unexplored for more complex scene-level point clouds and detection tasks.

To tackle the above challenges and exploit the potential of Mamba [10] in 3D scene understanding, in this paper, we propose 3DET-Mamba, an end-to-end 3D detector that fully takes advantage of Mamba. To effectively extract scene representations from unordered point cloud sequences, we introduce a local-to-global scanning technique that can capture local geometry as well as global representation. Specifically, the local-to-global scanning technique utilizes the Inner Mamba block to capture finer details in each local patch and then uses Dual Mamba blocks to further extract scene features in a global view. Additionally, given that the naive Mamba block struggles to effectively model the relationship between object queries and scene features, we propose a query-aware Mamba block that decodes scene context information into object sets more effectively, guided by box queries. Extensive experiments on standard benchmarks show that our method can outperform 3DETR [26] on ScanNet and SUN RGB-D. Moreover, the performance can be further improved by increasing the input point cloud and learnable query sequences.

Our contribution can be summarized as follows:

- We introduce Mamba into 3D scene perception for the first time and construct an end-to-end detector named 3DET-Mamba which fully takes advantage of Mamba.
- We design a local-to-global scanning mechanism and develop the Inner Mamba and Dual Mamba, which account for both local detailed features and global spatial representations, respectively. Further, we propose a Query-aware Mamba to decode scene context features through learnable queries and generate bounding boxes for objects of interest.
- Extensive experiments demonstrate that 3DET-Mamba outperforms previous 3DETR on both the ScanNet and SUN RGB-D datasets, proving that Mamba can serve as a promising foundational component for 3D scene understanding in the future.

## 2 Related Work

In this section, we will briefly introduce existing works on 1) 3D object detection, and 2) state space models and Mamba, especially Mamba in vision tasks.

### 2.1 3D Object Detection

With the development of robotics, 3D object detection [31, 9, 2, 51, 52] from point clouds has attracted increasing attention. Existing works can be divided into grid-based and point-based methods. Grid-based (voxel-based) methods convert the irregular point clouds into 2D grids [17] or 3D voxels [42, 35, 36, 59, 50] to utilize the strong feature extraction ability of CNN. Among them, FCAF3D [35] proposes a fully convolution anchor-free framework. CAGroup3D [42] designs a two-stage pipeline by introducing class-aware proposal generation and RoI-conv pooling. However, these grid-based methods struggle to utilize the inherent sparsity of the data and incur significant computational expenses due to the 3D convolution operations [31]. Inspired by PointNet family [32, 33], Point-based methods [37, 26, 24, 31, 57, 48, 5] directly generate 3D proposals from raw points. As a pioneer, VoteNet [31] proposes a voting mechanism to generate accurate proposals by generating new points close to the objects' center. RBGNet [43] design a ray-based representation which largely improves the performance. Motivated by the success of Transformer on image detection, 3DETR [26] and Groupfree [24] introduce a transformer-based architecture and boost the detection accuracy. However, the quadratic complexity of the attention mechanism struggles to model long-range dependencies with limited computational resources. In this paper, we present a novel detection framework that can serve as a promising foundational module in 3D detection.

### 2.2 State Space Models and Mamba

**State Space Models and Mamba.** Inspired by the success in control theory, state space model (SSM) [13, 38, 28, 12] has been utilized to model long-range dependence. Structured State-Space Sequence (S4) [13] replaces CNN and Transformer with SSM in vision and language tasks by

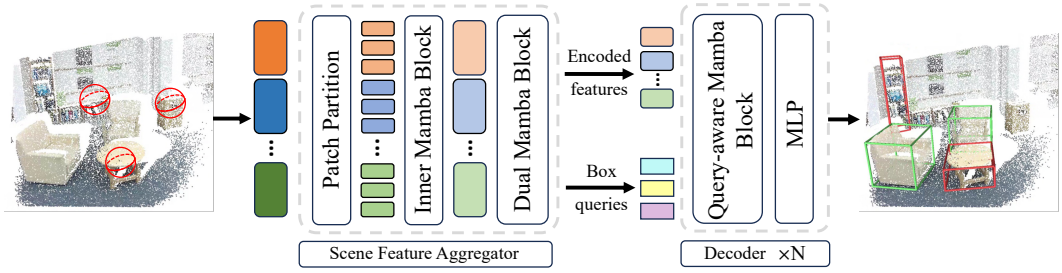


Figure 1: **The overview of the proposed 3DET-Mamba**. The point clouds are first patched and fed into the Inner Mamba block to learn fine-grained local features, which are then sent to the Dual Mamba block to extract global representations. These encoded scene information andd box queries go through the decoder, which includes Query-aware Mamba blocks and MLPs to generate the final bounding boxes. We employ the bipartite graph to match the predicted boxes with the ground truth and use a set loss for end-to-end optimization. Color is utilized only for visualization purposes.

combining linear SSM and HiPPO [11] framework. Smith *et al.* [38] propose S5 layer which designs a multi-input, multi-output SSM which outperforms S4 layer in both performance and efficiency. More recently, Mamba [10] with selective SSM is introduced which achieves higher performance than Transformer and leads to lots of further research on SSM [30, 44, 41]. For example, MoE-Mamba [30] and GraphMamba [41] combine Mamba with MoE and Graph data, respectively.

**Mamba in Vision.** Thanks to the breakthrough in natural language processing, Mamba has been rapidly transferred to various vision tasks [60, 23]. Vim[60] proposes a bi-directional SSM that can efficiently compress the vision representation and achieve satisfactory results on multiple vision tasks at low cost. VMamba [23] employs a cross-scan module to enable 1D selective scanning in 2D image space. Mamba-Unet [47] and MedMamba [53] introduce Mamba to medical image segmentation and classification, respectively. Video Mamba Suite [1] and VideoMamba [18] verify the effectiveness and efficiency of Mamba in various video tasks. QueryMamba [58] combines a query-based transformer decoder and the Mamba encoder to handle video action forecasting tasks. TM-Mamba [46] modifies the Mamba parameters as the function of the input and text query to ground the human motion. More recently, several works [54, 22, 21, 15] have explored the feasibility of Mamba on 3D tasks by introducing different point cloud ordering strategies. For example, PointMamba [21] utilizes Mamba to model the global information of 3D point clouds through a reordering mechanism and largely reduces the computation cost. Despite the great success achieved in object-level classification and part segmentation, Mamba in 3D scenes is under-explored. In this paper, we propose 3DET-Mamba which fills the gap of Mamba in 3D scene perception with designs like local-to-global scanning, dual Mamba, and query-aware Mamba.

## 3 Method

The overview framework of 3DET-Mamba is shown in Fig. 1. To better illustrate 3DET-Mamba, we first briefly review SSMs in Sec. 3.1. Then, we provide an overview of our 3DET-Mamba in Sec. 3.2. In Sec. 3.3 and Sec. 3.4, we detail the design of the encoder and decoder, respectively.

### 3.1 Preliminaries: State Space Models

State Space Models (SSMs) are derived from continuous systems and have been widely used in sequence modeling recently. Through a hidden state $h(t)$, SSMs can efficiently map 1D sequences input $x(t)$ to the output $y(t)$ using the following equations:

$$h'(t) = \mathbf{A}h(t) + \mathbf{B}x(t), \quad y(t) = \mathbf{C}h(t), \tag{1}$$

where $\mathbf{A} \in \mathbb{R}$ denotes the evolution matrix, and $\mathbf{B} \in \mathbb{R}^{N\times 1}$, $\mathbf{C} \in \mathbb{R}^{1\times N}$ are the projection matrices. To make SSMs can handle discrete signals such as language and point clouds, S4 [13] and Mamba [10] use a timescale parameter $\mathbf{\Delta}$ to transform the continuous parameters $\mathbf{A}$ and $\mathbf{B}$ into discrete ones $\overline{\mathbf{A}}$


https://doi.org/10.52202/079017-1497

and $\overline{\mathbf{B}}$. Specifically, Mamba uses zero-order hold (ZOH) method as follows:

$$\overline{\mathbf{A}} = \exp(\mathbf{\Delta A}), \quad \overline{\mathbf{B}} = (\mathbf{\Delta A})^{-1}(\exp(\mathbf{\Delta A}) - \mathbf{I}) \cdot \mathbf{\Delta B}. \tag{2}$$

Then, the Eq. (1) can be reformulated to the discrete version as follows:

$$h_t = \overline{\mathbf{A}} h_{t-1} + \overline{\mathbf{B}} x_t, \quad y_t = \mathbf{C} h_t. \tag{3}$$

Finally, a global convolution can be used to perform the model's calculation:

$$\overline{\mathbf{K}} = (\mathbf{C}\overline{\mathbf{B}}, \mathbf{C}\overline{\mathbf{A}\mathbf{B}}, ..., \mathbf{C}\overline{\mathbf{A}}^{L-1}\overline{\mathbf{B}}), \quad \mathbf{y} = \mathbf{x} * \overline{\mathbf{K}}, \tag{4}$$

where $L$ denotes the length of input sequence and $\overline{\mathbf{K}} \in \mathbb{R}^L$ presents a convolution kernel.

### 3.2 Overview

As shown in Fig. 1, our 3DET-Mamba takes a set of $N$ points $S \in \mathbb{R}^{N \times d}$ as input and generates a set of 3D (oriented) bounding boxes with semantic labels for all objects of interest. Specifically, 3DET-Mamba mainly consists of the Mamba-based scene feature aggregator and decoder designed for 3D object detection. The 3D feature aggregator combines Inner Mamba and Dual Mamba blocks to perform local-to-global scanning, extracting both the fine-grained local geometries and global contexts within the scene. On the decoding side, the decoder employs query-aware Mamba and Multi-Layer Perceptrons (MLPs) to decode the aggregated 3D features into discrete sets of objects. In the following sections, we will detail the designs of each component.

### 3.3 Scene Feature Aggregator

Recent advancements in point cloud processing have demonstrated the importance of capturing both local geometric features and global scene information [25]. However, previous methods primarily focus on global modeling using state-space models [21, 22, 54], resulting in a lack of fine-grained details. To tackle this challenge, we propose a novel local-to-global scanning technique that emphasizes scanning locally to uncover these finer details in each patch, followed by global scanning to capture the dependencies among local features.

In this section, we introduce a novel Mamba-based scene feature aggregator designed for local-to-global scanning, which is aimed at learning detailed representations of complex 3D scenes, as illustrated in Fig. 2. Specifically, our scene feature aggregator includes the Inner Mamba block and Dual Mamba block, which will introduced in Sec. 3.3.1 and Sec. 3.3.2 respectively.

#### 3.3.1 Inner Mamba

Given an unordered $N$ point cloud sequence $\{x_1, x_2, \ldots, x_n\}$, where $x_i \in \mathbb{R}^d$ and $d$ is the dimension of features, farthest point sampling (FPS) is first used to select $K$ key points from the input point cloud. Then k-nearest neighbors (KNN) is utilized to identify $N_0$ nearest neighbors for each key point, subsequently forming $K$ patches $P \in \mathbb{R}^{K \times N_0 \times d}$.

Previous works [33, 21, 22] employ MLPs to learn features from each patch, they struggle with the effective aggregation of local features. To address this, we treat the aggregation of local features as a sequence-to-sequence generation process and use a causal Mamba model to extract local features within a patch. Specifically, points within each patch are first normalized and then sorted based on their distance from the key point. These sequential points are fed into a lightweight Mamba block, of which the dimensions are reduced compared to the original Mamba block, to generate a new feature sequence. Finally, max-pooling is used to obtain the embedding for each patch, denoted as $F^{\text{s}} \in \mathbb{R}^{K \times C}$, where $K$ denotes the number of patches and $C$ is the embedding dimension.

#### 3.3.2 Dual Mamba

To encode point cloud data, previous works such as PointMamba [21] propose learning 3D representations by utilizing the original Mamba, which is initially designed for 1-D ordered sequences. Since understanding 3D data requires capturing global information, Point Cloud Mamba [54] suggests reversing the order of tokens and employing both forward and backward State Space Models (SSMs) to better capture the global context. However, due to the unordered and irregular nature of point

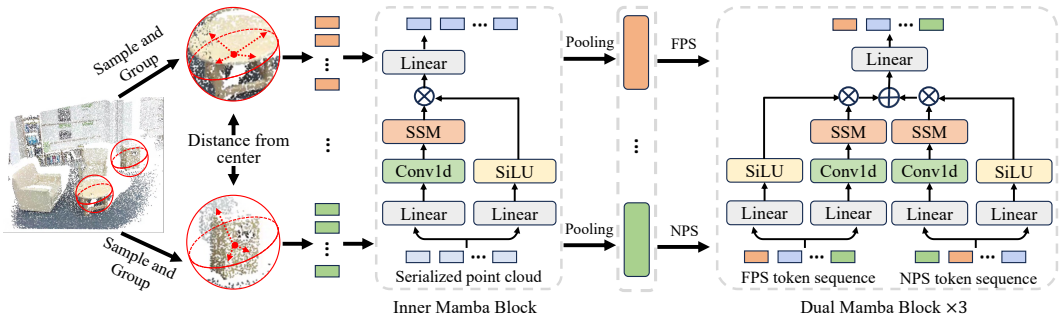


Figure 2: **Architecture of our scene feature aggregator that employs a novel local-to-global scanning mechanism.** The raw point clouds are first sampled and patched using FPS and KNN. Within each patch, points are ranked by their distance from the patch center. The Inner Mamba block then scans these ranked points to extract local geometric features. Subsequently, patches are treated as tokens and serialized in two manners before being fed into the Dual Mamba block. This step scans all tokens, extracting comprehensive scene contexts.

**Algorithm 1** Dual Mamba Block

**Input:** token sequence $\mathbf{T}_{l-1} : (\mathtt{B}, \mathtt{K}, \mathtt{C})$
**Output:** token sequence $\mathbf{T}_{l} : (\mathtt{B}, \mathtt{K}, \mathtt{C})$

/* process with different point orders */
$\mathbf{T}^{F}_{l-1} : (\mathtt{B}, \mathtt{K}, \mathtt{C}) \leftarrow \mathbf{T}_{l-1}$
$\mathbf{T}^{N}_{l-1} : (\mathtt{B}, \mathtt{K}, \mathtt{C}) \leftarrow \mathbf{NPS}(\mathbf{T}_{l-1})$
**for** $o$ in $\{F, N\}$ **do**
  $\mathbf{T}^{o}_{l-1}{}' \leftarrow \mathbf{Norm}(\mathbf{T}^{o}_{l-1})$
  $\mathbf{z}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{Linear}_{\mathbf{z}_o}(\mathbf{T}^{o}_{l-1}{}')$
  $\mathbf{x}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{Linear}_{\mathbf{x}_o}(\mathbf{T}^{o}_{l-1}{}')$
  $\mathbf{x}'_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{SiLU}(\mathbf{Conv1d}_o(\mathbf{x}_o))$
  $\mathbf{B}_o : (\mathtt{B}, \mathtt{K}, \mathtt{D}) \leftarrow \mathbf{Linear}^{\mathbf{B}}_{o}(\mathbf{x}'_o)$
  $\mathbf{C}_o : (\mathtt{B}, \mathtt{K}, \mathtt{D}) \leftarrow \mathbf{Linear}^{\mathbf{C}}_{o}(\mathbf{x}'_o)$
  /* softplus ensures positive $\mathbf{\Delta}_o$ */
  $\mathbf{\Delta}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \log(1 + \exp(\mathbf{Linear}^{\mathbf{\Delta}}_{o}(\mathbf{x}'_o) + \mathbf{Parameter}^{\mathbf{\Delta}}_{o}))$
  /* shape of $\mathbf{Parameter}^{\mathbf{A}}_{o}$ is $(\mathtt{C}', \mathtt{D})$ */
  $\overline{\mathbf{A}}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}', \mathtt{D}), \overline{\mathbf{B}}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}', \mathtt{D}) \leftarrow \mathbf{Disc}\,(\mathbf{\Delta}_o, \mathbf{Parameter}^{\mathbf{A}}_{o}, \mathbf{B}_o\,)$
  $\mathbf{y}_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{SSM}(\overline{\mathbf{A}}_o, \overline{\mathbf{B}}_o, \mathbf{C}_o)(\mathbf{x}'_o)$
  /* get gated $\mathbf{y}_o$ */
  $\mathbf{y}'_o : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{y}_o \bigodot \mathbf{SiLU}(\mathbf{z}_o)$
**end for**
$\mathbf{y}'_N : (\mathtt{B}, \mathtt{K}, \mathtt{C}') \leftarrow \mathbf{FPS}(\mathbf{y}'_N)$
$\mathbf{T}_l : (\mathtt{B}, \mathtt{K}, \mathtt{C}) \leftarrow \mathbf{Linear}^{\mathbf{T}}(\mathbf{y}'_F + \mathbf{y}'_N) + \mathbf{T}_{l-1}$
Return: $\mathbf{T}_l$

clouds, simply reversing the order of tokens cannot ensure the causal dependency of the point cloud sequence and may lead to unreliable results.

To mitigate such a problem, we introduce a novel Dual Mamba block (as shown in Fig. 2) to model long-range dependencies from the global view. Specifically, we treat $F^{\mathrm{s}} \in \mathbb{R}^{K \times C}$ as tokens and sort them based on their coordinates into two categories: farthest and nearest neighbor orders. The former strategy enhances the model's perception of spatial distribution by maximizing the distances between adjacent points in the sequence, whereas the latter ensures that adjacent points remain spatial neighbors, thus maintaining local consistency. The Dual Mamba block incorporates two branches to handle FPS and NPS token sequences, which first undergo normalization and independent linear projection. For each sequence, an initial 1D convolution transforms it into $\mathbf{x}'_o$, which is then projected


https://doi.org/10.52202/079017-1497

into $\mathbf{A}_o$, $\mathbf{B}_o$, and $\mathbf{\Delta}_o$. Subsequently, $\mathbf{A}_o$ and $\mathbf{B}_o$ are discretized using $\mathbf{\Delta}_o$. Finally, the corresponding tokens from the two branches are added and passed through a linear layer to generate the scene representations. The specific details are shown in Algorithm 1.

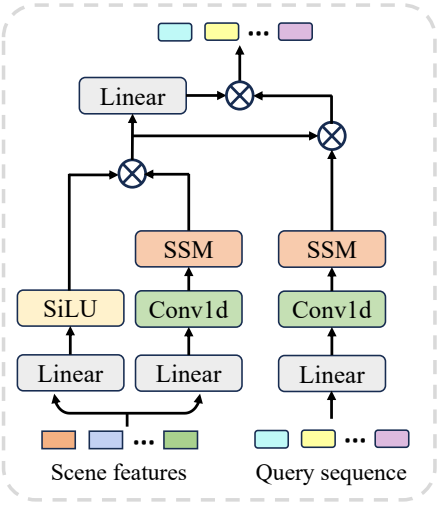


Figure 3: **Query-aware Mamba block.**

**Algorithm 2** Query-aware Mamba Block

**Input:** 3D scene features $\mathbf{F}^{\mathbf{s}}_{\mathbf{t-1}}$: $(\mathtt{B}, \mathtt{K}, \mathtt{C})$,
Box query sequence $\mathbf{F}^{\mathbf{q}}_{\mathbf{t-1}}$: $(\mathtt{B}, \mathtt{M}, \mathtt{C})$
**Output:** Query sequence $\mathbf{F}^{\mathbf{q}}_{\mathbf{t}}$: $(\mathtt{B}, \mathtt{M}, \mathtt{C})$

**for** $\mathbf{o}$ **in** $\{\mathbf{s}, \mathbf{q}\}$ **do**
  ${F^o_{t-1}}' \leftarrow \mathbf{Norm}(F^o_{t-1})$
  $\mathbf{x_o}$: $(\mathtt{B}, -, \mathtt{C}') \leftarrow \mathbf{Linear_{x_o}}(\mathbf{F^o_{t-1}}')$
  $\mathbf{x'_o}$: $(\mathtt{B}, -, \mathtt{C}') \leftarrow \mathbf{SiLu}(\mathbf{Conv_o}(\mathbf{x_o}))$
  /* Disc and SSM */
  $\overline{\mathbf{A}}$: $(\mathtt{B}, -, \mathtt{C}', \mathtt{D})$, $\overline{\mathbf{B}}$: $(\mathtt{B}, -, \mathtt{C}', \mathtt{D})$,
  $\mathbf{C}$ :$(\mathtt{B}, -, \mathtt{C}', \mathtt{D}) \leftarrow \mathbf{Disc}(\mathbf{x'_o})$
  $\mathbf{y_o}$: $(\mathtt{B}, -, \mathtt{C}') \leftarrow \mathbf{SSM}(\overline{\mathbf{A}}, \overline{\mathbf{B}}, \mathbf{C})(\mathbf{x'_o})$
  $\mathbf{z}$ :$(\mathtt{B}, -, \mathtt{C}') \leftarrow \mathbf{Linear_z}(\mathbf{F^s_{t-1}})$
  $\mathbf{y'_o}$: $(\mathtt{B}, -, \mathtt{C}') \leftarrow \mathbf{y_o} \odot \mathbf{SiLu}(\mathbf{z})$
**end for**
$\mathbf{y'_q}$: $(\mathtt{B}, \mathtt{M}, \mathtt{K}) \leftarrow \mathbf{y_q} \odot \mathbf{y'_s}$
$\mathbf{F^q_t}$: $(\mathtt{B}, \mathtt{M}, \mathtt{C}) \leftarrow \mathbf{Linear_F}(\mathbf{y'_s}) \odot \mathbf{y'_q}$
$\mathbf{F^q_t}$: $(\mathtt{B}, \mathtt{M}, \mathtt{C}) \leftarrow \mathbf{F^q_t} + \mathbf{Norm}(\mathbf{F^q_{t-1}})$
**return** $\mathbf{F^q_t}$;

### 3.4 Decoder

DETR-based models leverage a set of object queries to extract features for object classification and localization. However, directly using scene context as a prefix and concatenating it with queries in the Mamba model leads to suboptimal performance, as it struggles to capture discriminative features for independent queries. To address this, we introduce the Query-aware Mamba block for the decoder, as shown in Fig. 3, which effectively models the relationship between learnable queries and scene features to generate bounding boxes.

We first generate object queries by selecting a defined number of $M$ points from the set of $K$ key points using FPS, ensuring these query points cover the entire scene. For each of these points, we follow 3DETR [26] to transform their spatial coordinates into positional embeddings using the Fourier Transform. These embeddings are subsequently processed through an MLP to produce the initialized query embeddings.

In detail, the decoder is composed of $D$ same blocks and each block consists of a query-aware Mamba block and multiple MLP layers. The Query-aware Mamba block takes box queries and scene context as input, extracting tasked-related features from the scene context guided by the learnable queries. Specifically, each query sequence $\mathbf{F^q}$ is fed into a standard Mamba block to model the dependencies between queries. This process can be formulated as follows:

$$\begin{aligned} \mathbf{F^q_o} &= \mathbf{Linear}(\mathbf{Norm}(\mathbf{F^q})) \\ \mathbf{F^q} &= \mathbf{SiLU}(\mathbf{F^q_o}) \times \mathbf{SSM}(\mathbf{Conv}(\mathbf{F^q_o})). \end{aligned} \tag{5}$$

Meanwhile, scene features undergo the same process as the query sequence. Then, by multiplying the scene features with query embeddings, scene contexts are integrated into the query embeddings, and the updated queries are then passed through multiple MLP layers. After $D$ blocks decoding process, the bounding boxes and semantic categories can be generated using MLP-based heads. Please refer to the Algorithm 2 for more details.

### 3.5 Training Objectives

We adopt the same training objectives as [26] to train 3DET-Mamba. Specifically, we use the bipartite graph to match the set of predicted 3D bounding boxes $\{\hat{b}\}$ with the ground truth boxes $\{b\}$, denoted

https://doi.org/10.52202/079017-1497

as $L_{\text{giou}}(\hat{b}, b)$. Then we calculate the discrepancies between $\{\hat{b}\}$ and $\{b\}$ using $\ell_1$ loss for centers and dimensions, and Huber loss for angular residuals:

$$L_{\text{geo}} = \lambda_c \|\hat{c} - c\|_1 + \lambda_d \|\hat{d} - d\|_1 + \lambda_a \|\hat{a}_r - a_r\|_{\text{huber}}, \tag{6}$$

here, $\lambda_c$, $\lambda_d$,$\lambda_a$ are set as 5, 1, and 0.5. We also employ cross-entropy losses to assess angular and semantic classifications:

$$L_{\text{sem}} = -\lambda ac a_c \log \hat{a}_c - \lambda_s s \log \hat{s}, \tag{7}$$

here, $\lambda_{ac}$ and $\lambda_s$ are set as 0.1 and 1. Finally, the total loss is formulated as:

$$L_{\text{3DET-Mamba}} = L_{\text{giou}} + L_{\text{geom}} + L_{\text{sem}}. \tag{8}$$

# 4 Experiments

In this section, we first describe our experiment setups, including our benchmark datasets, metrics, and implementation details in Sec. 4.1. Then we present our main results in Sec. 4.2 and take out ablation studies to analyze the effectiveness of the proposed component in Sec. 4.3. Finally, we showcase some visualization results in Section Sec. 4.4.

## 4.1 Datasets, Metrics, and Implementation Details

**Datasets.** Following previous works on 3D indoor object detection, we evaluate our models on two challenging benchmarks: SUN RGB-D [39] and ScanNet [7]. The SUN RGB-D [39] dataset consists of 10,335 single-view RGB-D scans, with 5,285 used for training and 5,050 for validation. Each sample is annotated with rotated 3D bounding boxes. Following [31, 26], we convert the RGB-D images into point clouds using camera parameters and evaluate models on the 10 most common object categories. ScanNet [7] comprises 1,201 training samples and 312 validation samples, with each sample annotated with axis-aligned bounding box labels for 18 object categories.

**Metrics.** Following [24, 26], we use standard evaluation protocols [31, 35] and report the detection performance on the validation set using mean Average Precision (mAP) at two different IoU thresholds (*i.e.*m 0.25 and 0.5), denoted as mAP@0.25 and mAP@0.5, respectively.

**Implementation Details.** The input to our model is a point cloud $P \in \mathbb{R}^{N\times 3}$ representing a 3D scene, with $N$ set as 20,000 for SUN RGB-D [39] and 40,000 for ScanNet [7]. We employ a single-layer inner mamba block that generates 2048 patches, each with 256-dimensional features. The dual mamba encoder has 3 layers and outputs scene features with a hidden dimension of 256. The decoder has 8 layers and is closely followed by MLPs as the bounding box prediction head. During training, we employ standard data augmentation methods, including random cropping, sampling, and flipping. We use the AdamW optimizer with a base learning rate of $7 \times 10^{-4}$, decayed to $10^{-6}$ using a cosine schedule, and a weight decay of 0.1. Gradient clipping with an $\ell_2$ norm of 0.1 is applied to stabilize training. The whole model is implemented in PyTorch, and all experiments are conducted on 8 NVIDIA 3090 GPUs (24 GB) with a total batch size of 64.

## 4.2 Comparisons on 3D Object Detection

In this section, we compare our 3DET-Mamba with previous 3D detectors. As shown in Tab. 1, our 3DET-Mamba can outperform previous 3DETR [26] (Tramsformer-based detectors) on both SUN RGB-D and ScanNet datasets. For example, with 256 queries and 2048 points, our method achieves 66.9% mAP@0.25 and 48.7% mAP@0.5 on ScanNet, surpassing 3DETR-m, which obtains 65.0% mAP@0.25 and 47.0% mAP@0.5 respectively. Besides, since Mamba can effectively handle long sequences, we further conduct experiments using point clouds with higher resolution and more box queries (*i.e.*, 4096 point clouds and 512 box queries), we can observe that the performance can be further improved with longer point and query sequences (*i.e.*, +5.7 mAP@0.5).

## 4.3 Ablation Studies

**Analysis of the encoder.** To verify the effectiveness of the designed Mamba-based encoder, we first conduct ablation studies on Inner Mamba and Dual Mamba. Specifically, we replace Inner Mamba and Dual Mamba with Pointnet++ and Transformer. As reported in Tab. 2, transforming Inner


https://doi.org/10.52202/079017-1497

Table 1: 3D detection results on ScanNet V2 [7] and SUN RGB-D [39]. We compare 3DET-Mamba against open-source methods that directly process point clouds, using PointNet++[33] (PN) and Transformer[40] (Tran.) as backbones. 3DET-Mamba employs the same number of key points and queries as 3DETR [26]. 3DET-Mamba† doubles the number of key points and queries to model longer sequences of point clouds. Compared to 3DETR [26], 3DET-Mamba achieves superior performance.

| Methods | Backbone | ScanNet | | SUN RGB-D | |
|---|---|---|---|---|---|
| | | mAP@0.25 | mAP@0.5 | mAP@0.25 | mAP@0.5 |
| VoteNet [31] | PN | 58.6 | 33.5 | 57.7 | - |
| MLCVNet [49] | | 64.5 | 41.4 | 59.8 | - |
| H3DNet [56] | | 67.2 | 48.1 | 60.1 | 39.0 |
| BRNet [6] | | 66.1 | 50.9 | 61.1 | 43.7 |
| GroupFree3D [24] | | 67.3 | 48.9 | 63.0 | 45.2 |
| GroupFree3D* [24] | | 69.1 | 52.8 | - | - |
| 3DETR [27] | Tran. | 62.7 | 37.5 | 58.0 | 30.3 |
| 3DETR-m [27] | | 65.0 | 47.0 | 59.1 | 32.7 |
| 3DET-Mamba | Mamba | 66.9 | 48.7 | 59.3 | 33.4 |
| 3DET-Mamba† | | **70.4** | **54.4** | **61.3** | **42.2** |

Table 2: Effect of our designed point cloud encoder which contains Inner and Dual Mamba blocks. We compare the performance of encoder combinations using Pointnet++ [33] and Transformer [26] with Inner and Dual Mamba blocks, with results showing that the proposed Inner Mamba & Dual Mamba achieves the best performance.

| Encoder | mAP@0.25 | mAP@0.5 |
|---|---|---|
| Pointnet++ & Transformer | 63.1 | 44.1 |
| Inner Mamba & Transformer | 64.9 | 45.4 |
| Pointnet++ & Dual Mamba | 65.6 | 48.4 |
| Inner Mamba & Dual Mamba | **66.9** | **48.7** |

Mamba to PointNet++ results in a performance drop of 1.8% mAP@0.25 and 1.3% mAP@0.5 when using Transformer and Dual Mamba, respectively. This is because our Inner Mamba can effectively aggregate and propagate local features. Besides, our Dual Mamba block can bring +4.3% mAP@0.5 and +3.3% mAP@0.5 improvement compared to using Transformer as the spatial encoder since our Mamba layers can learn both spatial representation and local consistency at the same time. By combining the Inner and Dual Mamba, the best results can be achieved which further verifies our designs.

**Effect of Dual Mamba block.** To further show the advantage of our designed Dual Mamba block, we replace the Dual Mamba block with the original Mamba [10] block and Bi-Mamba block [60, 22, 54], respectively. The original Mamba block only contains a forward SSM and the Bi-Mamba block is composed of a forward and a backward SSM. As shown in Tab. 3, our proposed dual Mamba block can outperform both the original Mamba block and Bi-Mamba block by +1.0% and +1.2% mAP@0.5. This is because our dual Mamba block can provide point cloud sequence with short-range and long-range dependencies and further make better use of the powerful causal modeling ability of Mamba.

**Effect of Query-aware Mamba.** We conduct ablation studies on the decoder to demonstrate the effectiveness of the designed Query-aware Mamba block. We compare our module against the following baselines: (a) a transformer-based decoder as proposed in 3DETR [26]; (b) a naive approach that directly concatenates 3D scene information with queries, feeding them into the Mamba block and leveraging Mamba's state transition matrix for further modeling; and (c) our proposed Query-aware Mamba block. Additionally, all experiments are conducted with the same encoder and training strategy to ensure a fair comparison. As demonstrated in Tab. 4, the decoder based on the Query-aware Mamba block achieves superior results compared to both baselines. This improvement can be attributed to two main factors: (1) Directly concatenating queries and 3D scene representations do

https://doi.org/10.52202/079017-1497

Table 3: Effect of Dual Mamba block. We compare it with the original Mamba block [10] and the Bi-directional Mamba block [60, 22].

| Mamba Block | mAP@0.25 | mAP@0.5 |
|---|---|---|
| Ori. Mamba | 65.4 | 47.7 |
| Bi. Mamba | 66.4 | 47.5 |
| Dual Mamba | **66.9** | **48.7** |

Table 4: Effect of Query-aware Mamba. We compare it with the 3DETR [26] decoder and an implementation based on the original Mamba.

| Decoder | mAP@0.25 | mAP@0.5 |
|---|---|---|
| Transformer | 64.3 | 42.6 |
| Ori. Mamba | 56.6 | 28.0 |
| Quety-aware Mamba | **66.9** | **48.7** |

not effectively model the relationship between them, making it difficult to capture the most relevant 3D features for each query. (2) Our Query-aware Mamba decoder more effectively aggregates dependencies between queries and scene features, enhancing the extraction of task-relevant features from the context.

### 4.4 Qualitative Results.

To demonstrate the effectiveness of 3DET-Mamba more intuitively, we show some visualization results. As shown in Fig. 4, we can observe that our 3DET-Mamba can accurately detect objects.

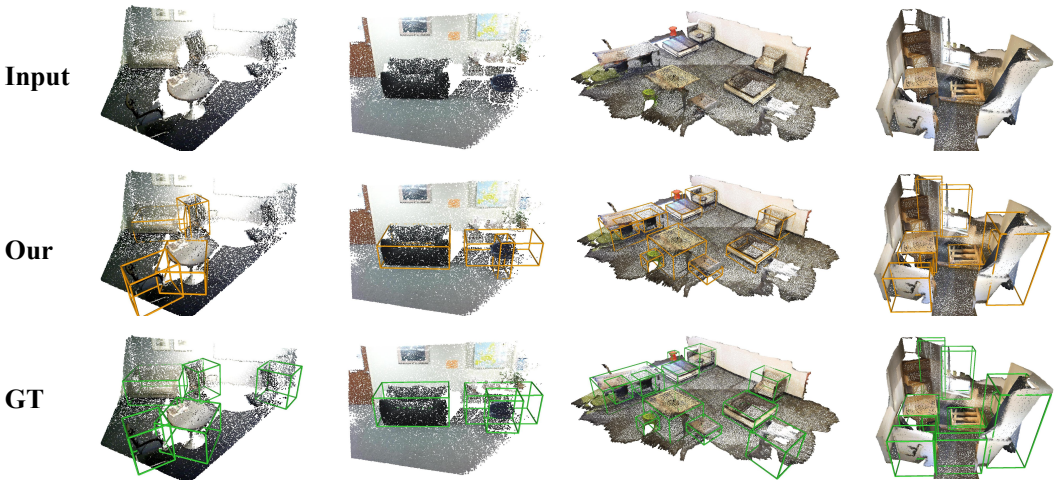


Figure 4: **Visualization of detection results.** 3DET-Mamba is able to generate tight bounding boxes for objects of interest in these complex and diverse scenes.

## 5 Conclusion

In this paper, for the first time, we exploit the potential of Mamba in 3D object detection tasks. Specifically, we introduce 3DET-Mamba, an end-to-end detector based on encoder-decoder architecture. We first propose an SSM-based encoder that uses an Inner Mamba block to capture local geometric information and uses Dual Mamba blocks to further aggregate features in a global view. Besides, a Query-aware Mamba module is designed to effectively decode scene representations into object sets with the guide of learnable box queries. Extensive experiments on standard benchmarks like ScanNet and SUN RGB-D demonstrate the effectiveness of 3DET-Mamba, proving Mamba as a promising building block for future 3D scene understanding. In addition, with the increased length of input point sequence and query sequence, the performance will be further boosted since Mamba is an expert in long sequence modeling.

## 6 Limitations and Broader Impacts

While 3DET-Mamba has demonstrated effectiveness in modeling point cloud sequences, we have yet to explore its potential for handling other 3D data types, such as meshes. Additionally, given Mamba's


https://doi.org/10.52202/079017-1497

strength in modeling long-sequence data, a promising future direction is to develop Mamba-based 3D foundation models capable of addressing a broader range of scene-level tasks, including 3D dense caption, visual grounding, and 3D QA. We leave these explorations for future work.

## Acknowledgement

This work is supported by National Natural Science Foundation of China (No. 62071127, and 62101137), National Key Research and Development Program of China (No. 2022ZD0160101), Shanghai Natural Science Foundation (No. 23ZR1402900), Shanghai Municipal Science and Technology Major Project (No.2021SHZDZX0103). The computations in this research were performed using the CFFF platform of Fudan University.

## References


[1] G. Chen, Y. Huang, J. Xu, B. Pei, Z. Chen, Z. Li, J. Wang, K. Li, T. Lu, and L. Wang. Video mamba suite: State space model as a versatile alternative for video understanding. *arXiv preprint arXiv:2403.09626*, 2024.

[2] J. Chen, B. Lei, Q. Song, H. Ying, D. Z. Chen, and J. Wu. A hierarchical graph network for 3d object detection on point clouds. In *CVPR*, pages 389–398, 2020.

[3] S. Chen, X. Chen, C. Zhang, M. Li, G. Yu, H. Fei, H. Zhu, J. Fan, and T. Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning. *arXiv preprint arXiv:2311.18651*, 2023.

[4] S. Chen, H. Zhu, X. Chen, Y. Lei, G. Yu, and T. Chen. End-to-end 3d dense captioning with vote2cap-detr. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11124–11133, 2023.

[5] S. Chen, H. Zhu, M. Li, X. Chen, P. Guo, Y. Lei, G. Yu, T. Li, and T. Chen. Vote2cap-detr++: Decoupling localization and describing for end-to-end 3d dense captioning. *arXiv preprint arXiv:2309.02999*, 2023.

[6] B. Cheng, L. Sheng, S. Shi, M. Yang, and D. Xu. Back-tracing representative points for voting-based 3d object detection in point clouds. In *CVPR*, pages 8963–8972, 2021.

[7] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, pages 5828–5839, 2017.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[9] F. Engelmann, M. Bokeloh, A. Fathi, B. Leibe, and M. Nießner. 3d-mpa: Multi-proposal aggregation for 3d semantic instance segmentation. In *CVPR*, pages 9028–9037.

[10] A. Gu and T. Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[11] A. Gu, T. Dao, S. Ermon, A. Rudra, and C. Ré. Hippo: Recurrent memory with optimal polynomial projections. In *NeurIPS*, 2020.

[12] A. Gu, K. Goel, A. Gupta, and C. Ré. On the parameterization and initialization of diagonal state space models. In *NeurIPS*, 2022.

[13] A. Gu, K. Goel, and C. Re. Efficiently modeling long sequences with structured state spaces. In *ICLR*, 2021.

[14] T. Guan, J. Wang, S. Lan, R. Chandra, Z. Wu, L. Davis, and D. Manocha. M3detr: Multi-representation, multi-scale, mutual-relation 3d object detection with transformers. In *wACV*, pages 772–782, 2022.

[15] X. Han, Y. Tang, Z. Wang, and X. Li. Mamba3d: Enhancing local features for 3d point cloud analysis via state space model. *arXiv preprint arXiv:2404.14966*, 2024.

[16] M. M. Islam and G. Bertasius. Long movie clip classification with state-space video models. In *ECCV*, 2022.

[17] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, pages 12697–12705, 2019.

[18] K. Li, X. Li, Y. Wang, Y. He, Y. Wang, L. Wang, and Y. Qiao. Videomamba: State space model for efficient video understanding. *arXiv preprint arXiv:2403.06977*, 2024.

[19] M. Li, X. Chen, C. Zhang, S. Chen, H. Zhu, F. Yin, G. Yu, and T. Chen. M3dbench: Let's instruct large models with multi-modal 3d prompts. *arXiv preprint arXiv:2312.10763*, 2023.

[20] M. Li, L. Zhang, M. Zhu, Z. Huang, G. Yu, J. Fan, and T. Chen. Lightweight model pre-training via language guided knowledge distillation. *IEEE Transactions on Multimedia*, 2024.

[21] D. Liang, X. Zhou, X. Wang, X. Zhu, W. Xu, Z. Zou, X. Ye, and X. Bai. Pointmamba: A simple state space model for point cloud analysis. *arXiv preprint arXiv:2402.10739*, 2024.



https://doi.org/10.52202/079017-1497

[22] J. Liu, R. Yu, Y. Wang, Y. Zheng, T. Deng, W. Ye, and H. Wang. Point mamba: A novel point cloud backbone based on state space model with octree-based ordering strategy. *arXiv preprint arXiv:2403.06467*, 2024.

[23] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, and Y. Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024.

[24] Z. Liu, Z. Zhang, Y. Cao, H. Hu, and X. Tong. Group-free 3d object detection via transformers. In *CVPR*, pages 2949–2958, 2021.

[25] X. Ma, C. Qin, H. You, H. Ran, and Y. Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022.

[26] I. Misra, R. Girdhar, and A. Joulin. An End-to-End Transformer Model for 3D Object Detection. In *ICCV*, 2021.

[27] I. Misra, R. Girdhar, and A. Joulin. An end-to-end transformer model for 3d object detection. In *CVPR*, pages 2906–2917, 2021.

[28] E. Nguyen, K. Goel, A. Gu, G. Downs, P. Shah, T. Dao, S. Baccus, and C. Ré. S4nd: Modeling images and videos as multidimensional signals with state spaces. In *NeurIPS*, 2022.

[29] J. Park, S. Lee, S. Kim, Y. Xiong, and H. J. Kim. Self-positioning point-based transformer for point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21814–21823, 2023.

[30] M. Pióro, K. Ciebiera, K. Król, J. Ludziejewski, and S. Jaszczur. Moe-mamba: Efficient selective state space models with mixture of experts. *arXiv preprint arXiv:2401.04081*, 2024.

[31] C. R. Qi, O. Litany, K. He, and L. J. Guibas. Deep hough voting for 3d object detection in point clouds. In *ICCV*, pages 9276–9285, 2019.

[32] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, pages 652–660, 2017.

[33] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, pages 5099–5108, 2017.

[34] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

[35] D. Rukhovich, A. Vorontsova, and A. Konushin. Fcaf3d: fully convolutional anchor-free 3d object detection. In *ECCV*, pages 477–493. Springer, 2022.

[36] D. Rukhovich, A. Vorontsova, and A. Konushin. Tr3d: Towards real-time indoor 3d object detection. pages 281–285, 2023.

[37] S. Shi, X. Wang, and H. Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, pages 770–779, 2019.

[38] J. T. Smith, A. Warrington, and S. Linderman. Simplified state space layers for sequence modeling. In *ICLR*, 2022.

[39] S. Song, S. P. Lichtenberg, and J. Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, pages 567–576, 2015.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[41] C. Wang, O. Tsepa, J. Ma, and B. Wang. Graph-mamba: Towards long-range graph sequence modeling with selective state spaces. *arXiv preprint arXiv:2402.00789*, 2024.

[42] H. Wang, L. Ding, S. Dong, S. Shi, A. Li, J. Li, Z. Li, and L. Wang. Cagroup3d: Class-aware grouping for 3d object detection on point clouds. *arXiv preprint arXiv:2210.04264*, 2022.

[43] H. Wang, S. Shi, Z. Yang, R. Fang, Q. Qian, H. Li, B. Schiele, and L. Wang. Rbgnet: Ray-based grouping for 3d object detection. In *CVPR*, pages 1100–1109, 2022.

[44] J. Wang, T. Gangavarapu, J. N. Yan, and A. M. Rush. Mambabyte: Token-free selective state space model. *arXiv preprint arXiv:2401.13660*, 2024.

[45] J. Wang, W. Zhu, P. Wang, X. Yu, L. Liu, M. Omar, and R. Hamid. Selective structured state-spaces for long-form video understanding. In *CVPR*, 2023.

[46] X. Wang, Z. Kang, and Y. Mu. Text-controlled motion mamba: Text-instructed temporal grounding of human motion. *arXiv preprint arXiv:2404.11375*, 2024.

[47] Z. Wang, J.-Q. Zheng, Y. Zhang, G. Cui, and L. Li. Mamba-unet: Unet-like pure visual mamba for medical image segmentation. *arXiv preprint arXiv:2402.05079*, 2024.

[48] Q. Xie, Y. Lai, J. Wu, Z. Wang, D. Lu, M. Wei, and J. Wang. Venet: Voting enhancement network for 3d object detection. In *ICCV*, pages 3692–3701.

[49] Q. Xie, Y.-K. Lai, J. Wu, Z. Wang, Y. Zhang, K. Xu, and J. Wang. Mlcvnet: Multi-level context votenet for 3d object detection. In *CVPR*, pages 10447–10456, 2020.

[50] Y. Yan, Y. Mao, and B. Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[51] J. Yuan, B. Zhang, X. Yan, T. Chen, B. Shi, Y. Li, and Y. Qiao. Bi3d: Bi-domain active learning for cross-domain 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15599–15608, 2023.

[52] J. Yuan, B. Zhang, X. Yan, B. Shi, T. Chen, Y. Li, and Y. Qiao. Ad-pt: Autonomous driving pre-training with large-scale point cloud dataset. *Advances in Neural Information Processing Systems*, 36, 2024.

[53] Y. Yue and Z. Li. Medmamba: Vision mamba for medical image classification. *arXiv preprint arXiv:2403.03849*, 2024.

[54] T. Zhang, X. Li, H. Yuan, S. Ji, and S. Yan. Point could mamba: Point cloud learning via state space model. *arXiv preprint arXiv:2403.00762*, 2024.

[55] Z. Zhang, A. Liu, I. Reid, R. Hartley, B. Zhuang, and H. Tang. Motion mamba: Efficient and long sequence motion generation with hierarchical and bidirectional selective ssm. *arXiv preprint arXiv:2403.07487*, 2024.

[56] Z. Zhang, B. Sun, H. Yang, and Q. Huang. H3dnet: 3d object detection using hybrid geometric primitives. In *ECCV*, pages 311–329, 2020.

[57] Y. Zheng, Y. Duan, J. Lu, J. Zhou, and Q. Tian. Hyperdet3d: Learning a scene-conditioned 3d object detector. In *CVPR*, pages 5575–5584.

[58] Z. Zhong, M. Martin, F. Diederichs, and J. Beyerer. Querymamba: A mamba-based encoder-decoder architecture with a statistical verb-noun interaction module for video action forecasting@ ego4d long-term action anticipation challenge 2024. *arXiv preprint arXiv:2407.04184*, 2024.

[59] Y. Zhou and O. Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, pages 4490–4499, 2018.

[60] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, and X. Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.

 https://doi.org/10.52202/079017-1497

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in our abstract and introduction that our 3DET-Mamba can boost previous 3DETR with the help of our novel SSM-based architecture accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Detailed information needed to reproduce the results are illustrated by algorithms and texts in Section 3

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**


https://doi.org/10.52202/079017-1497

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Data and code are not included in the submission due to the time limit, we will release our code in the future.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We specify them in Section 4

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The results do not contain error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Compute resources are reported in Section 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research is conducted with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.


https://doi.org/10.52202/079017-1497

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original papers or websites that produced the code package or dataset.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.


https://doi.org/10.52202/079017-1497